

ADVANCE JAVASCRIPT

MODULE: 1 (Introduction and Code Quality)

1. Write a program to Show an alert

```
⇒ <!DOCTYPE html>

<html>

<head>

  <title>Alert Example</title>

</head>

<body>

<script>

  alert("This is an alert!");

</script>

</body>

</html>
```

2. What will be the result for these expressions?

1. $5 > 4 \rightarrow \text{true}$
2. $\text{"apple"} > \text{"pineapple"} \rightarrow \text{false}$
3. $\text{"2"} > \text{"12"} \rightarrow \text{true}$

4. `undefined == null` → true

5. `undefined === null` → false

6. `null == "\n0\n"` → false

7. `null === +"\n0\n"` → false

3. Will alert be shown? `if ("0") { alert('Hello'); }`

→ yes

4. What is the code below going to output? `alert(null || 2 || undefined);`

→ output is 2

5. The following function returns true if the parameter age is greater than 18. Otherwise it asks for a confirmation and returns its result:

function

`checkAge(age)`

`{`

`if (age > 18) { return true; }`

`else`

`{ // ...return confirm ('did parents allow you?'); }`

→ `function checkAge(age) {`

`if (age > 18) {`

```
    return true;
  } else {
    return confirm("Did parents allow you?");
  }
}
```

6. Replace Function Expressions with arrow functions in the code below: `Function ask(question, yes, no) { if (confirm(question))yes(); else no(); } ask("Do you agree?", function() { alert("You agreed."); }, function() { alert("You canceled the execution."); } }`

```
➔const ask = (question, yes, no) => {
  if (confirm(question)) {
    yes();
  } else {
    no();
  }
};
```

```
ask("Do you agree?", () => {
  alert("You agreed.");
}, () => {
  alert("You canceled the execution.");
});
```

});

MODULE: 2 (Data Types and Objects)

1. Write the code, one line for each action:

a) Create an empty object user.

➔ `let user = {};`

b) Add the property name with the value John.

➔ `user.name = "meet";`

c) Add the property surname with the value Smith.

➔ `user.surname = "panchal";`

d) Change the value of the name to Pete.

➔ `user.name = "meet";`

e) Remove the property name from the object.

➔ `delete user.name;`

2. Is array copied? `let fruits = ["Apples", "Pear", "Orange"]; // push a new value into the "copy" let shoppingCart = fruits; shoppingCart.push("Banana"); // what's in fruits? alert(fruits.length); // ?`

➔ No, the array is not copied. When you do `shoppingCart = fruits`, you are not creating a new array; instead, you are creating a reference to the same array in memory. Therefore, any changes made to `shoppingCart` will also affect the `fruits` array.

3. Map to names

```
let john = { name: "John", age: 25 }; let pete = { name: "Pete", age: 30 };  
let mary = { name: "Mary", age: 28 }; let users = [ john, pete, mary ];  
let names = /* ... your code */ alert( names ); // John, Pete, Mary
```

➔

```
let john = { name: "maical", surname: "jordan", id: 1 };
```

```
let pete = { name: "ethen", surname: "Hunt", id: 2 };
```

```
let mary = { name: "mical", surname: "jordan", id: 3 };
```

```
let users = [maical,ethen,michal];
```

```
let names = users.map(user => user.name);
```

```
alert(names);
```

4. Map to objects

```
let john = { name: "John", surname: "Smith", id: 1 };  
let pete = { name: "Pete", surname: "Hunt", id: 2 }; let mary = { name: "Mary",  
surname: "Key", id: 3 }; let users = [ john, pete, mary ]; let  
usersMapped = /* ... your code ... */
```

➔

```
let john = { name: "maical", surname: "jordan", id: 1 };
```

```
let pete = { name: "ethen", surname: "Hunt", id: 2 };
```

```
let mary = { name: "mical", surname: "jordan", id: 3 };
```

```
let users = [john, pete, mary];
```

```
let usersMapped = users.map(user => {  
  return {  
    fullName: `${user.name} ${user.surname}`,  
    id: user.id  
  };  
});
```

```
console.log(usersMapped);
```

MODULE: 3 (Document, Event and Controls)

1. Create a program to hide/show the password

➔ `<input type="password" class="password-input" id="password">`
`<button id="togglePassword">Toggle Password</button>`

`<script>`

```
const passwordInput = document.getElementById('password');  
const toggleButton = document.getElementById('togglePassword');
```

```
toggleButton.addEventListener('click', () => {  
  if (passwordInput.type === 'password') {  
    passwordInput.type = 'text';  
  } else {  
    passwordInput.type = 'password';  
  }  
});
```

`</script>`

2. Create a program that will select all the classes and loop over and whenever i click the button the alert should show

➔ `<!-- HTML -->`


```
<button id="showAlertButton">Show Alert</button>
```

```
<div class="myClass">Element 1</div>
```

```
<div class="myClass">Element 2</div>
```

```
<div class="myClass">Element 3</div>
```

```
<div class="myClass">Element 4</div>
```

```
<script>
```

```
document.getElementById('showAlertButton').addEventListener('click', function () {
```

```
    const elementsWithClass = document.querySelectorAll('.myClass');
```

```
    elementsWithClass.forEach(element => {
```

```
        alert(element.textContent); // Show alert with the content of each element
```

```
    });
```

```
});
```

```
</script>
```

3. Create a responsive header using proper JavaScript

➔ <!DOCTYPE html>

```
<html>
```

```
<head>
```

<style>

/* CSS for styling the header */

.header {

background-color: #333;

color: #fff;

padding: 10px;

display: flex;

justify-content: space-between;

align-items: center;

}

.logo {

font-size: 24px;

font-weight: bold;

}

.nav {

display: flex;

list-style: none;

margin: 0;

```
padding: 0;  
}
```

```
.nav li {  
    margin-right: 20px;  
}
```

```
/* CSS for mobile navigation */  
.mobile-nav {  
    display: none;  
}
```

```
@media (max-width: 768px) {  
    .nav {  
        display: none;  
    }
```

```
.mobile-nav {  
    display: block;  
}
```

}

</style>

</head>

<body>

<div class="header">

<div class="logo">Your Logo</div>

<ul class="nav">

Home

About

Services

Contact

<div class="mobile-nav">

<button id="toggleMobileNav">Menu</button>

<ul id="mobileNavList" class="nav">

Home

About

Services

Contact

</div>

</div>

<script>

// JavaScript for responsive header

**const toggleMobileNavButton =
document.getElementById('toggleMobileNav');**

const mobileNavList = document.getElementById('mobileNavList');

toggleMobileNavButton.addEventListener('click', function() {

if (mobileNavList.style.display === 'block') {

mobileNavList.style.display = 'none';

} else {

mobileNavList.style.display = 'block';

}

});

</script>

</body>

</html>

